IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages

Divyanshu Kakwani[1], **Anoop Kunchukuttan**[2]*, **Satish Golla**[3*], **Gokul N.C.**[4], **Avik Bhattacharyya**[5], **Mitesh M. Khapra**[6], **Pratyush Kumar**[7]
Robert Bosch Centre for Data Science and AI, IIT Madras[1,6,7], Microsoft India[2], AI4Bharat[3,4,5]
{divk,miteshk,pratyush}@cse.iitm.ac.in[1,6,7],
ankunchu@microsoft.com[2], gokulnc@ai4bharat.org[4],
{gsatishkumaryadav,avikbhattacharyya.2k}@gmail.com[3,5]

Abstract

In this paper, we introduce NLP resources for 11 major Indian languages from two major language families. These resources include: (a) large-scale sentence-level monolingual corpora, (b) pre-trained word embeddings (c) pre-trained language models and (d) multiple NLU evaluation datasets (*IndicGLUE* benchmark). The monolingual corpora contains a total of 8.9 billion tokens across all 11 languages and Indian English, primarily sourced from news crawls. The word embeddings are based on *FastText*, hence suitable for handling morphological complexity of Indian languages. The pre-trained language models are based on the compact ALBERT model. Lastly, the *IndicGLUE* benchmark for Indian language NLU contains datasets for the following tasks: Article Genre Classification, Headline Prediction, Named Entity Recognition, Cross-lingual Sentence Retrieval, Wikipedia Section-Title Prediction and Cloze-style Multiple choice QA. Our embeddings are competitive or better than existing pre-trained embeddings on multiple tasks. We hope that the availability of the dataset will accelerate Indic NLP research which has the potential to impact more than a billion people. It can also help the community in evaluating advances in NLP over a more diverse pool of languages. The data and models can be found at https://indicnlp.ai4bharat.org

1 Introduction

Distributional representations are the corner stone of modern NLP, which have led to significant advances in many NLP tasks like text classification, NER, sentiment analysis, MT, QA, NLI, *etc.* Particularly, word embeddings (Mikolov et al., 2013b), contextualized word embeddings (Peters et al., 2018), and language models (Devlin et al., 2019) can model syntactic/semantic relations between words and reduce feature engineering. These pre-trained models are useful for initialization and/or transfer learning for NLP tasks. They are also useful for learning multilingual embeddings which enable cross-lingual transfer. Pre-trained models are typically learned from large, diverse monolingual corpora. The quality of embeddings is impacted by the size of the monolingual corpora (Mikolov et al., 2013a; Bojanowski et al., 2017), a resource not widely available for many major languages.

In particular, Indic languages, widely spoken by more than a billion speakers, lack large, publicly available monolingual corpora. They include 8 out of top 20 most spoken languages and ~30 languages with more than a million speakers. There is also a growing population of users consuming Indian language content (print, digital, government and businesses). Further, Indic languages are very diverse, spanning 4 major language families. The Indo-Aryan and Dravidian languages are spoken by 96% of the population in India. The other families are diverse, but the speaker population is relatively small. Almost all Indian languages have SOV word order and are morphologically rich. The language families have also interacted over a long period of time leading to significant convergence in linguistic features; hence, the Indian subcontinent is referred to as a *linguistic area* (Emeneau, 1956). Indic languages are thus of great interest and importance for NLP research.

Unfortunately, the progress on Indic NLP has been constrained by the unavailability of large scale monolingual corpora and evaluation benchmarks. The former allows the development of pre-trained language models and deep contextualised word embeddings which have become drivers of modern NLP. The latter allows systematic evaluation across a wide variety of tasks to check the efficacy of new models. With the hope of accelerating Indic

*Volunteer effort for the AI4Bharat project

NLP research, we address the creation of (i) large, general-domain monolingual corpora for multiple Indian languages, (ii) word embeddings and multilingual language models trained on this corpora, and (iii) an evaluation benchmark comprising of various NLU tasks.

Our monolingual corpora, collectively referred to as *IndicCorp*, contain a total of 8.9 billion tokens across 11 major Indian languages and English. The data in *IndicCorp* are primarily sourced from news crawls. Using *IndicCorp*, we first train and evaluate word embeddings for each of the 11 languages. Given the morphological richness of Indian languages we train FastText word embeddings which are known to be more effective for such languages. To evaluate these embeddings we curate a benchmark comprising of word similarity and analogy tasks (Akhtar et al., 2017; Grave et al., 2018), text classification tasks, sentence classification tasks (Akhtar et al., 2016; Mukku and Mamidi, 2017), and bilingual lexicon induction tasks. The key finding is that on most tasks the word embeddings trained on our *IndicCorp* outperform similar embeddings trained on existing corpora for Indian languages.

Next, we train multilingual language models for these 11 languages using the ALBERT model (Lan et al., 2019). We chose ALBERT as the base model as it is very compact and hence easier to use in downstream tasks. To evaluate these pre-trained language models, we create an NLU benchmark comprising of the following tasks: article genre classification, headline prediction, named entity recognition, Wikipedia section-title prediction, cloze-style multiple choice QA and cross-lingual sentence retrieval. Across all these tasks, we show that our embeddings are competitive or better than existing pre-trained multilingual embeddings such as mBERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2019). We hope that these embeddings and evaluations benchmarks will not only be useful in driving NLP research on Indic languages, but will also help in evaluating advances in NLP over a more diverse set of languages.

In summary, this paper contributes introduces *IndicNLPSuite* containing the following resources for Indic NLP which will be made publicly available:

- *IndicCorp*: Large sentence-level monolingual corpora for 11 languages from two language families (Indo-Aryan branch and Dravidian) and Indian English with an average 9-fold increase in size over OSCAR.
- *IndicFT* and *IndicBERT*: FastText-based word emebeddings and ALBERT-based language models for 11 languages trained on *IndicCorp*. The *IndicBERT* embeddings are multilingual and also support English trained on Indian English news sources.
- *IndicGLUE*: An evaluation benchmark containing a variety of NLU tasks.

2 Related Work

Text Corpora. Few organized sources of monolingual corpora exist for most Indian languages. The EMILLE/CIIL corpus (McEnery et al., 2000) was an early effort to build corpora for South Asian languages, spanning 14 languages with a total of 92 million words. *Wikipedia* for Indian languages is small (the largest one, Hindi, has just 40 million words). The Leipzig corpus (Goldhahn et al., 2012) contains small collections of upto 1 million sentences for news and web crawls (average 300K sentences). In addition, there are some language specific corpora for Hindi and Urdu (Bojar et al., 2014; Jawaid et al., 2014). In particular, the HindMonoCorp (Bojar et al., 2014) is one of the few larger Indian language collections (787 million token Hindi corpus).

The *CommonCrawl*[1] project crawls webpages in many languages by sampling various websites. Our analysis of a processed crawl for the years 2013-2016 (Buck et al., 2014) for Indian languages revealed that most Indian languages, with the exception of Hindi, Tamil and Malayalam, have few good sentences (≥10 words) - in the order of around 50 million words. The OSCAR project (Ortiz Suarez et al., 2019), a recent processing of CommonCrawl, also contains much less data for most Indian languages than our crawls. The CCNet () and C4 () projects also provide tools to process common crawl, but the extracted corpora are not provided and require a large amount of processing power. Our monolingual corpora si about 4 times larger than the corresponding OSCAR corpus and two times larger than the corresponding CC-100 corpus ().

Word Embeddings. Word embeddings have been trained for many Indian languages using limited corpora. The Polyglot (Al-Rfou et al., 2013) and FastText (Bojanowski et al., 2017) projects provide embeddings trained on Wikipedia. FastText also

[1] https://commoncrawl.org

Language		#S	#T	#V	I/O
Punjabi	(pa)	24.2	814	3.0	22
Hindi	(hi)	56.8	1,840	6.5	2
Bengali	(bn)	37.3	815	6.6	2
Odia	(or)	6.2	104	1.4	9
Assamese	(as)	1.0	36.9	0.8	8
Gujarati	(gu)	35.8	724	5.7	14
Marathi	(mr)	30.8	560	5.8	7
Kannada	(kn)	46.3	712	11.9	14
Telugu	(te)	43.3	671	9.4	8
Malayalam	(ml)	50.6	767	17.7	8
Tamil	(ta)	29	549	11.4	2
English	(en)	47.3	1,341	4.5	
Total		408.6	8,934	84.7	

Table 1: *IndicCorp* de-duplicated monolingual corpora statistics: number of sentences (S), tokens (T), types (V) in millions, the ratio of *IndicCorp* size to OSCAR corpus size (I/O)

provides embeddings trained on Wikipedia + CommonCrawl corpora. We show that on most evaluation tasks *IndicFT* outperforms existing FastText based embeddings for Indian languages.

Pretrained Transformers. Pre-trained transformers serve as general language understanding models that can be used in a wide variety of downstream NLP tasks (Radford et al., 2019). Several transformer-based language models such as GPT(Radford, 2018), BERT(Devlin et al., 2019) and its variants like RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), *etc.* have been proposed. All these models require large amounts of monolingual corpora for training. For Indic languages, two such multilingual models are available: XLM-R (Conneau et al., 2019) and multilingual BERT (Devlin et al., 2019). However, they are trained across multiple languages and on much smaller Indic language corpora.

NLU Benchmarks. Benchmarks such as GLUE (Wang et al., 2018), SuperGLUE (Wang et al., 2019), CLUE (Chinese) (Xu et al., 2020), and FLUE (French) (Le et al., 2019) are important for tracking the efficacy of NLP models across languages. Such a benchmark is missing for Indic languages and the goal of this work is to fill this void.

3 *IndicCorp*: Indian language corpora

In this section, we describe the creation of our monolingual corpora.

Data sources. Our goal is collection of corpora that reflect contemporary use of Indic languages and cover a wide range of topics. Hence, we focus primarily on crawling news articles, magazines and blogposts. We source our data from popular Indian language news websites. We discover most of our sources through online newspaper directories (*e.g.*, w3newspaper) and automated web searches using hand-picked terms in various languages.

We analyzed whether we could augment our crawls with data from other smaller sources like Leipzig corpus (Goldhahn et al., 2012), WMT NewsCrawl, WMT CommonCrawl (Buck et al., 2014), HindEnCorp (Hindi) (Bojar et al., 2014), *etc*. Amongst these we chose to augment our dataset with only the CommonCrawl data from the OSCAR corpus (Ortiz Suárez et al., 2019).

Article Extraction. For many news websites, we used *BoilerPipe*[2], a tool to automatically extract the main article content for structured pages without any site-specific customizations (Kohlschütter et al., 2010). This approach works well for most of the Indian language news websites. In some cases, we wrote custom extractors for each website using *BeautifulSoup*[3], a Python library for parsing HTML/XML documents. After content extraction, we applied filters on content length, script, *etc.*, to select good quality articles.

Text Processing. First, we canonicalize the representation of Indic language text in order to handle multiple Unicode representations of certain characters. Next, we split the article into sentences and tokenize the sentences. These steps take into account Indic punctuations and sentence delimiters. Heuristics avoid creating sentences for initials (P. G. Wodehouse) and common Indian titles (Shri., equivalent to Mr. in English) which are followed by a period. We use the *Indic NLP Library*[4] (Kunchukuttan, 2020) for processing.

The final corpus for a language is created after combining our crawls with OSCAR corpus[5] and de-duplicating and shuffling sentences. We used the Murmurhash algorithm (*mmh3* Python library with a 128-bit unsigned hash) for de-duplication. Due to copyright reasons, we only release the final shuffled corpus described below.

Dataset Statistics. Table 1 shows statistics of the de-duplicated monolingual datasets for each language. Hindi and Indian English are the largest collections, while Odia and Assamese have the smallest collection. All other languages have a

[2] https://github.com/kohlschutter/boilerpipe
[3] https://www.crummy.com/software/BeautifulSoup
[4] https://github.com/anoopkunchukuttan/indic_nlp_library
[5] https://oscar-corpus.com/

Lang	FT-W	FT-WC	*IndicFT*
Word Similarity (*Pearson Correlation*)			
pa	**0.467**	0.384	0.445
hi	0.575	0.551	**0.598**
gu	0.507	0.521	**0.600**
mr	0.497	**0.544**	0.509
te	0.559	0.543	**0.578**
ta	**0.439**	0.438	0.422
Average	0.507	0.497	**0.525**
Word Analogy (*% accuracy*)			
hi	19.76	**32.93**	29.65

Table 2: Word Similarity and Analogy Results for different pre-trained embeddings. (a) **FT-W**: FastText Wikipedia, (b) **FT-WC**: FastText Wikipedia + CommonCrawl, (c) ***IndicFT***: IndicNLP.

collection between 500-1000 million words. OSCAR is an important contributor to our corpus and accounts for nearly (23%) of our corpus by the number of sentences. The rest of the data originate from our crawls. As evident from the last column of Table 1, for 8 languages the number of tokens in our corpus is at least 7 times that in OSCAR. For the remaining 3 languages it is twice that of OSCAR.

4 *IndicFT*: Indian Language Word Embeddings

We train FastText word embeddings for each language using *IndicCorp*, and evaluate their quality on: (a) word similarity, (b) word analogy, (c) text classification, (d) bilingual lexicon induction tasks. We compare our embeddings (referred to as *IndicFT*) with two pre-trained embeddings released by the *FastText* project trained on Wikipedia (**FT-W**) (Bojanowski et al., 2017) and Wiki+CommonCrawl (**FT-WC**) (Grave et al., 2018) respectively.

4.1 Training Details

We train 300-dimensional word embeddings for each language on *IndicCorp* using *FastText* (Bojanowski et al., 2017). Since Indian languages are morphologically rich, we chose *FastText*, which is capable of integrating subword information by using character n-gram embeddings during training. We train skipgram models for 10 epochs with a window size of 5, minimum token count of 5 and 10 negative examples sampled for each instance. We chose these hyper-parameters based on suggestions by Grave et al. (2018). Based on previously published results, we expect FastText to be better than word-level algorithms like *word2vec* (Mikolov et al., 2013b) and *GloVe* (Pennington et al., 2014) for morphologically rich languages.

4.2 Word Similarity & Analogy Evaluation

We perform an intrinsic evaluation of the word embeddings using the IIIT-Hyderabad word similarity dataset (Akhtar et al., 2017) which contains similarity databases for 7 Indian languages. The database contains similarity judgments for around 100-200 word-pairs per language. Table 2 shows the evaluation results. We also evaluated the Hindi word embeddings on the Facebook Hindi word analogy dataset (Grave et al., 2018). On average, *IndicFT* embeddings outperform the baseline embeddings.

4.3 Text Classification Evaluation

We evaluated the embeddings on different text classification tasks: (a) news article topic, (b) news headlines topic and (c) sentiment classification. We experimented on publicly available datasets and a new dataset (*IndicGLUE* News Category dataset).

Publicly available datasets. We used the following datasets: (a) IIT-Patna Sentiment Analysis dataset (Akhtar et al., 2016), (b) ACTSA Sentiment Analysis corpus (Mukku and Mamidi, 2017), (c) BBC News Articles classification dataset, (d) iNLTK Headlines dataset, and (e) Soham Bengali News classification dataset. (See Appendix A for dataset details). Our train and test splits derived from the above mentioned corpora will be made publicly available.

***IndicGLUE* News Category Dataset.** We use *IndicCorp* to create classification datasets comprising news articles and their categories for 9 languages. The categories are determined from URL components. We chose generic categories like entertainment and sports which are likely to be consistent across websites. The datasets are balanced across classes. Please refer to Table 6 and Appendix B for more details.

Classifier training. Following Meng et al. (2019), we use a k-NN ($k = 4$) classifier since it is nonparameteric. Hence, classification performance directly reflects how well the embedding space captures text semantics. The input text embedding is the mean of all word embeddings (bag-of-words assumption).

Results. On nearly all datasets and languages, *IndicFT* embeddings outperform baseline embeddings (see Tables 3 and 4).

Lang	Dataset	FT-W	FT-WC	*IndicFT*
hi	BBC Articles	72.29	67.44	**77.02**
	IITP+ Movie	41.61	44.52	**45.81**
	IITP Product	58.32	57.17	**61.57**
bn	Soham Articles	62.79	64.78	**71.82**
gu	iNLTK Headlines	81.94	84.07	**90.74**
ml		86.35	83.65	**95.87**
mr		83.06	81.65	**91.40**
ta		90.88	89.09	**95.37**
te	ACTSA	46.03	42.51	**52.58**
	Average	69.25	68.32	**75.80**

Table 3: Text classification accuracy on public datasets

Lang	FT-W	FT-WC	*IndicFT*
pa	**97.12**	95.53	96.47
bn	96.57	97.57	**97.71**
or	94.80	96.20	**98.43**
gu	95.12	94.63	**99.02**
mr	96.44	97.07	**99.37**
kn	95.93	96.53	**97.43**
te	98.67	98.08	**99.17**
ml	89.02	89.18	**92.83**
ta	95.99	95.90	**97.26**
Average	95.52	95.63	**97.52**

Table 4: Accuracy on our *IndicGLUE* News category testset

4.4 Bilingual Lexicon Induction

We use *IndicFT* embeddings for creating multilingual embeddings, where monolingual word embeddings from different languages are mapped into the same vector space. Cross-lingual learning using multilingual embeddings is useful for Indic languages which are related and where training data for NLP tasks is skewed across languages. We train bilingual word embeddings from English to Indian languages and vice versa using GeoMM (Jawanpuria et al., 2019), a state-of-the-art supervised method for learning bilingual embeddings. We evaluate the bilingual embeddings on the BLI task, using bilingual dictionaries from the MUSE project and *en-te* dictionary created in-house. We search among the 200k most frequent target language words with the CSLS distance metric during inference (Conneau et al., 2018). Table 5 shows the results. The quality of multilingual embeddings depends on the quality of monolingual embeddings. *IndicFT* bilingual embeddings significantly outperform the baseline bilingual embeddings for most languages.

	en to Indic			Indic to en		
	FT-W	FT-WC	*IndicFT*	FT-W	FT-WC	*IndicFT*
bn	22.60	33.92	**36.68**	31.22	42.10	**42.67**
hi	40.93	**44.35**	41.53	49.56	**57.16**	54.85
te	21.10	23.01	**51.11**	25.36	32.84	**57.58**
ta	19.27	30.25	**31.87**	26.66	**40.20**	38.65
Ave.	25.98	32.88	**40.29**	33.20	43.08	**48.38**

Table 5: Accuracy@1 for bilingual lexicon induction

5 *IndicGLUE*: Indian Language NLU Benchmark

We now introduce *IndicGLUE*, the Indic General Language Understanding Evaluation Benchmark, which is a collection of various tasks as described below. Table 6 summarises the sizes of the respective datasets. Further details (such as the min, max, average number of words per training instance) can be found in Appendix C.

Headline Prediction Task. The task is to predict the correct headline for a news article from a given list of four candidate headlines (3 incorrect, 1 correct). We generated the dataset for this task from our news article crawls which contain articles and their headlines. We ensured that the three incorrect candidates are not completely unrelated to the given article. In particular, while choosing incorrect candidates, we considered only those articles that had a sizeable overlap of entities with the original article. We used min-hash and locality-sensitive hashing to efficiently search such articles.

Named Entity Recognition. We use the publicly available data[6] by (Pan et al., 2017) which contains NER data for 282 languages. They created this data from Wikipedia by exploiting cross language links to propagate English named entity labels to other languages. For all our evaluations, we consider the following coarse-grained labels in this data: Person (PER), Organisation (ORG) and Location (LOC). The annotations are in the standard BIO notation.

Wikipedia Section-title Prediction. The task is to predict the correct title for a Wikipedia section from a given list of four candidate titles (3 incorrect, 1 correct). We use the open-source tool WikiExtractor to extract sections and their titles from Wikipedia. To increase the classification challenge, we choose the 3 incorrect candidates for a given section, only from the titles of other sections in the same article as the given section.

[6]https://elisa-ie.github.io/wikiann/

pa	hi	bn	or	as	gu	mr	kn	te	ml	ta	total
Headline Prediction											
100,000	100,000	68,350	100,000	49,751	100,000	67,571	56,457	63,415	100,000	74,767	880,311
Wikipedia Section-Title Prediction											
10,966	55,087	59,475	5,019	6,251	12,506	13,058	44,224	100,000	34,409	61,175	402,170
Named Entity Recognition											
9,462	69,431	109,508	8,687	6,295	39,708	108,579	28,854	81,627	138,888	186,423	787,462
News Category Classification											
3,120	-	14,000	30,000	-	2,040	4,770	30,000	24,000	6,000	11,700	125,630
Cloze-style QA											
5,664	35,135	38,845	1,975	2,942	22,856	11,370	13,656	41,338	26,531	38,585	238,897
Cross-lingual Sentence Retrieval (#English to Indian language parallel sentences)											
-	5,169	5,522	752	-	6,463	5,760	-	5,049	4,886	5,637	39,238

Table 6: *IndicGLUE* Datasets' Statistics

Cloze-style Multiple-choice QA. Given a text with an entity randomly masked, the task is to predict that masked entity from a list of 4 candidate entities (3 incorrect, 1 correct). The text is obtained from Wikipedia articles and the entities in the text are identified using Wikidata. We choose the 3 incorrect candidates from entities that occur in the same article and have the same type as the correct entity. The type of an entity is taken from Wikidata. This task is similar to the one proposed in (Petroni et al., 2019) for English, and aims to check if language models can be used as knowledge bases.

News Category Classification. The task is to predict the genre of a given news article. We use the News Category Classification dataset that we proposed in Section 4.3. Recall that this dataset contains news articles and their categories for 9 languages (categories are: entertainment, sports, business, lifestyle, technology, politics, crime with balanced number of articles across categories).

Cross-lingual Sentence Retrieval. Given a sentence in language L_1 the task is to retrieve its translation from a set of candidate sentences in language L_2. We construct this corpus by filtering the *Mann Ki Baat dataset*[7] by IIIT-H CVIT (Siripragrada et al., 2020) to include only clean sentences for language pairs.

6 *IndicBERT*

In this section, we introduce *IndicBERT* which is trained on our monolingual corpora and then evaluated on *IndicGLUE*. We specifically chose ALBERT as the base model as it has a smaller parameter size making it easier to distribute and use in downstream applications. Further, similar to mBERT, we chose to train a single model for all Indian languages with a hope of exploiting the relatedness amongst Indian languages. In particular, such joint training may be beneficial for some of the under represented languages (e.g., Odia and Assamese).

6.1 Pre-training

Using *IndicCorp* we first train a sentence piece tokenizer (Kudo and Richardson, 2018) to tokenize the sentences in each language. We use this tokenized corpora to train a multilingual ALBERT using the standard masked language model (MLM) objective. Note that we did not use the Sentence Order Prediction objective used in the original ALBERT work. Similar to mBERT and XLM-R models, we perform exponentially smoothed weighting of the data across languages to give a better representation to low-resource languages. We choose a vocabulary of 200k to accommodate different scripts and large vocabularies of Indic languages.

We train our models on a single TPU v3 provided by Tensorflow Research Cloud (TFRC[8]). We train both the base and large versions of ALBERT. To account for memory constraints, we use a smaller maximum sequence length of 128. In addition, for the large model, we use a smaller batch size of 2048. For creating each batch, we first randomly select a language and then randomly select sentences from that language. Apart from sequence length

[7] http://preon.iiit.ac.in/ jerin/bhasha/

[8] https://www.tensorflow.org/tfrc

Model	pa	hi	bn	or	as	gu	mr	kn	te	ml	ta	avg
News Article Headline Prediction												
XLM-R	97.44	94.72	94.62	93.20	96.14	97.28	**94.79**	98.16	91.30	96.32	96.90	95.52
mBERT	94.32	94.56	90.64	52.64	92.92	94.24	90.77	96.88	88.40	94.24	95.72	89.58
IndicBERT base	97.36	95.36	**95.91**	**93.84**	96.62	97.36	93.85	97.88	89.16	**96.48**	96.26	95.46
IndicBERT large	**97.68**	**95.68**	95.79	93.28	**97.43**	**97.92**	93.14	**98.16**	**92.69**	95.20	**97.65**	**95.87**
Wikipedia Section Title Prediction												
XLM-R	70.29	76.92	80.91	68.25	56.96	27.39	77.44	24.41	94.64	**76.10**	76.34	66.33
mBERT	72.47	**80.12**	82.53	22.22	**73.42**	**74.52**	**80.49**	78.84	94.56	74.25	76.86	**73.66**
IndicBERT base	67.39	74.02	80.11	57.14	65.82	68.79	72.56	75.05	94.80	75.87	74.90	73.31
IndicBERT large	**77.54**	77.80	**82.66**	**68.25**	56.96	52.23	77.44	**80.11**	**95.36**	64.27	71.37	73.09
Cloze-style multiple-choice QA												
XLM-R	29.31	30.62	29.95	35.98	27.11	11.15	32.38	29.36	27.16	27.57	27.24	27.98
mBERT	33.70	39.00	36.23	26.37	29.42	**83.31**	38.81	33.96	**37.58**	**36.71**	**35.72**	39.16
IndicBERT base	**44.74**	**41.55**	**39.40**	**39.32**	**40.49**	70.78	**44.85**	**39.57**	32.60	35.39	31.83	**41.87**
IndicBERT large	41.91	37.01	32.63	33.81	30.03	52.73	39.98	32.28	26.73	28.04	28.10	34.84

Table 7: Test accuracy on various multiple-choice tasks

Model	Params	#Train Tokens Total	Indic
XLM-R	125M	295B	3.99B
mBERT	110M	18.2B*	184M*
IndicBERT base	12M	8.93B	7.59B
IndicBERT large	18M	8.93B	7.59B

Table 8: Comparison of Different Models. *Estimated

and batch size, the remaining hyperparameters are the default values as in Lan et al. (2019). We train the model for a total of 400k steps. It took 6 days to train the base model and 9 days to train the large model. In the remaining discussion, we refer to our models as *IndicBERT* base and *IndicBERT* large.

6.2 Fine-tuning

After pre-training, we fine-tune *IndicBERT* on each of the tasks in *IndicGLUE*. The fine-tuning is done independently for each task and each language (i.e., in the end we have a task-specific model for each language). We divide each dataset into a train set (80%), development set (10%), and test set (10%). We only use the train set for fine-tuning. Below, we describe the fine-tuning procedure followed for each task for both versions of the model (base and large). As a common hyperparameter, we fine-tuned the models for 3 epochs.

Headline Prediction Task. We feed the *article* and *candidate headline* to the model with a SEP token in between. We have a classification head at the top which assigns a score between 0 and 1 to the headline. We use cross entropy loss with the target label as 1 for the correct candidate and 0 for the incorrect candidates. During prediction, we choose the candidate headline which is assigned the highest score by our model.

Named Entity Recognition. Each sentence is fed as a single sequence to the model. For every token, we have a softmax layer at the output which computes a probability distribution over the NER classes (following the BIO convention). We fine-tune the model using multi-class cross entropy loss.

Wikipedia Section Title Prediction. We follow the same procedure as for the Headline Prediction Task (instead of a news article we have a Wikipedia section and instead of candidate headlines we have candidate titles).

Cloze-style Multiple-choice QA. We feed the masked text segment as input to the model and at the output we have a softmax layer which predicts a probability distribution over the given candidates. We fine-tune the model using cross entropy loss with the target label as 1 for the correct candidate and 0 for the incorrect candidates.

News Category Classification. We use the representation of the [CLS] token from the last layer as the representation of the input news article. We then feed this representation to a linear classifier with a softmax layer to predict a probability distribution over the genres. We fine-tune the model using multi-class cross entropy loss.

Cross-lingual Sentence Retrieval. No fine-tuning is required for this task. We compute the representation of every sentence by mean-pooling

Model	pa	hi	bn	or	as	gu	mr	kn	te	ml	ta	avg
Article Genre Classification												
XLM-R	94.87	-	**98.29**	97.07	-	96.15	96.67	97.60	99.33	**96.00**	**97.28**	97.03
mBERT	94.87	-	97.71	69.33	-	84.62	96.67	97.87	98.67	81.33	94.56	90.63
IndicBERT base	**97.44**	-	97.14	97.33	-	**100.00**	**96.67**	97.87	99.67	93.33	96.60	**97.34**
IndicBERT large	94.87	-	97.71	**97.60**	-	73.08	95.00	**97.87**	**99.67**	85.33	95.24	92.93
Named Entity Recognition (F1-score)												
XLM-R	17.86	89.62	**92.95**	25.00	66.67	55.32	87.86	47.06	81.71	81.98	79.16	65.93
mBERT	**50.00**	86.56	91.81	19.05	**92.31**	68.04	**91.27**	59.72	84.31	82.64	79.90	**73.24**
IndicBERT base	21.43	**90.30**	93.39	8.69	41.67	54.74	88.71	52.29	**84.38**	83.16	**90.45**	64.47
IndicBERT large	44.44	86.81	91.85	**35.09**	43.48	**70.21**	87.73	**63.51**	80.12	**84.35**	80.81	69.85

Table 9: Test accuracy on various classification tasks

Model	en-hi	en-bn	en-or	en-gu	en-mr	en-te	en-ml	en-ta	avg
XLM-R	4.77	9.46	15.96	18.46	18.07	15.23	17.47	10.48	13.74
mBERT	**33.73**	26.30	2.66	17.68	24.67	26.13	16.76	23.78	21.46
IndicBERT base	24.67	26.12	33.11	28.17	23.09	25.10	31.22	25.44	27.12
IndicBERT large	21.99	**29.00**	**49.60**	**39.43**	**32.67**	**34.30**	**32.26**	**33.58**	**34.10**

Table 10: Precision@10 on Cross-Lingual Sentence Retrieval Task

the outputs in the last hidden layer and then using cosine distance to compute similarity between sentences (Libovický et al., 2019). Additionally, we also center the sentence vectors across each language to remove language-specific bias in the vectors (Reimers and Gurevych, 2019).

6.3 Evaluation

Below we summarize the main observations from our results as reported in Tables 7 to 10.

Comparison with mBERT and XML-R. In 4 out of the 6 tasks, *IndicBERT* models outperform XLM-R and mBERT. Further, *IndicBERT* models are competitive on the Wikipedia Section Title prediction task, but are clearly out-performed by mBERT on the NER dataset.

Performance on Wikipedia Tasks. We notice that the performance of mBERT is relatively higher for the tasks based on Wikipedia data, namely NER, Wikipedia Section Title prediction, and Multiple-choice QA. This suggests that mBERT, unlike other models, is benefiting from exposure to Wikipedia data during its training. Note that we deliberately did not include Wikipedia in our monolingual corpora as it is a good source for creating NLU tasks (hence, to keep things clean we didn't want it to be a part of pre-training).

Small v/s Large *IndicBERT*. The large and base models of *IndicBERT* are comparable: There are two tasks each on which either model is clearly better, and two tasks on which both models perform similarly.

Challenging tasks. Multiple-choice QA and Cross-Lingual Sentence Retrieval prove to be the more challenging tasks. On both tasks, *IndicBERT* models improve on XLM-R and mBERT.

Effect of corpus size. Comparing across languages, on the 5 mono-lingual tasks the performance of *IndicBERT* large is poorest on Assamese and Odia, the two languages with the smallest corpora sizes (see Table 1). On the other hand, performance is highest on Hindi and Bengali, which have the largest corpora sizes (see Table 1). This reinforces the expectation that accuracy is sensitive to the corpora size.

7 Conclusion and Future Work

We present the *IndicNLPSuite* dataset, a collection of large-scale, general-domain, sentence-level corpora of 8.9 billion words across 11 Indian languages, along with *IndicFT*, *IndicBERT* and *IndicGLUE*. We show that resources derived from this dataset outperform other pre-trained embeddings on many NLP tasks. The sentence-level corpora, embeddings and evaluation datasets will be publicly available for research and non-commercial use under a *Creative Commons Attribution-NonCommercial-ShareAlike 4.0 Inter-*

national License.

In addition to building embeddings, *IndicNLP-Suite* can be useful for different NLP tasks like NMT backtranslation, unsupervised morphanalysis, parallel translation and transliteration corpus mining, *etc*. We hope the availability of these datasets will accelerate NLP research for Indian languages by enabling the community to build further resources and solutions for various NLP tasks and opening up interesting NLP questions.

References

Md. Shad Akhtar, Ayush Kumar, Asif Ekbal, and Pushpak Bhattacharyya. 2016. A hybrid deep learning architecture for sentiment analysis. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 482–493.

Syed Sarfaraz Akhtar, Arihant Gupta, Avijit Vajpayee, Arjit Srivastava, and Manish Shrivastava. 2017. Word similarity datasets for Indian languages: Annotation and baseline systems. In *Proceedings of the 11th Linguistic Annotation Workshop*, pages 91–94, Valencia, Spain. Association for Computational Linguistics.

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Ondrej Bojar, Vojtech Diatka, Pavel Rychlỳ, Pavel Stranák, Vít Suchomel, Ales Tamchyna, and Daniel Zeman. 2014. Hindencorp-hindi-english and hindi-only corpus for machine translation. In *LREC*, pages 3550–3555.

Christian Buck, Kenneth Heafield, and Bas van Ooyen. 2014. N-gram counts and language models from the common crawl. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3579–3584, Reykjavik, Iceland. European Language Resources Association (ELRA).

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *Proceedings of the International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Murray B Emeneau. 1956. India as a lingustic area. *Language*.

D. Goldhahn, T. Eckart, and U. Quasthoff. 2012. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2012)*.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Bushra Jawaid, Amir Kamran, and Ondrej Bojar. 2014. A Tagged Corpus and a Tagger for Urdu. In *LREC*, pages 2938–2943.

Pratik Jawanpuria, Arjun Balgovind, Anoop Kunchukuttan, and Bamdev Mishra. 2019. Learning multilingual word embeddings in latent metric space: a geometric approach. *Transaction of the Association for Computational Linguistics (TACL)*, 7:107–120.

Aditya Joshi, AR Balamurali, Pushpak Bhattacharyya, et al. 2010. A fall-back strategy for sentiment analysis in Hindi: a case study. In *Proceedings of the 8th ICON*.

Christian Kohlschütter, Péter Fankhauser, and Wolfgang Nejdl. 2010. Boilerplate detection using shallow text features. In *WSDM*.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Anoop Kunchukuttan. 2020. The IndicNLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. 2019. Flaubert: Unsupervised language model pre-training for french.

Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2019. How language-neutral is multilingual bert?

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Anthony McEnery, Paul Baker, Rob Gaizauskas, and Hamish Cunningham. 2000. Emille: Building a corpus of south asian languages. *VIVEK-BOMBAY-*, 13(3):22–28.

Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance Kaplan, and Jiawei Han. 2019. Spherical text embedding. In *Advances in Neural Information Processing Systems*, pages 8206–8215.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Sandeep Sricharan Mukku and Radhika Mamidi. 2017. ACTSA: Annotated corpus for Telugu sentiment analysis. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 54–58, Copenhagen, Denmark. Association for Computational Linguistics.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*, Cardiff, United Kingdom. Leibniz-Institut für Deutsche Sprache.

Pedro Javier Ortiz Suarez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora*, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Alec Radford. 2018. Improving language understanding by generative pre-training. `https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf`.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. `https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf`.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Shashank Siripragrada, Jerin Philip, Vinay P. Namboodiri, and C V Jawahar. 2020. A multilingual parallel corpora collection effort for Indian languages. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3743–3751, Marseille, France. European Language Resources Association.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 3266–3280. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Liang Xu, Xuanwei Zhang, Lu Li, Hai Hu, Chenjie Cao, Weitang Liu, Junyi Li, Yudong Li, Kai Sun, Yechen Xu, et al. 2020. Clue: A chinese language understanding evaluation benchmark. *arXiv preprint arXiv:2004.05986*.

A Publicly Available Text Classification Datasets

We used the following publicly available datasets for our text classification experiments:

(a) IIT-Patna Movie and Product review dataset (Akhtar et al., 2016), (b) ACTSA Sentiment Analysis corpus (Mukku and Mamidi, 2017), (c) IIT-Bombay Sentiment Analysis Dataset (Joshi et al., 2010), (d) BBC News Articles classification dataset, (e) iNLTK Headlines dataset, (f) Soham Bengali News classification corpus. The essential details of the datasets are described in Table 11.

Some notes on the above mentioned public datasets

- The IITP+ Movie Reviews sentiment analysis dataset is created by merging IIT-Patna dataset with the smaller IIT-Bombay and iNLTK datasets.
- The IIT-Patna Movie and Product review datasets have 4 classes namely postive, negative, neutral and conflict. We ignored the conflict class.
- In the Telugu-ACTSA corpus, we evaluated only on the news line dataset (named as telugu_sentiment_fasttext.txt) and ignored all the other domain datasets as they have very few data-points.

Lang	Dataset	N	# Examples Train	Test
hi	BBC Articles[9]	6	3,467	866
	IITP+ Movie Reviews	3	2,480	310
	IITP Product Reviews[10]	3	4,182	523
bn	Soham Articles[11]	6	11,284	1411
gu	iNLTK Headlines[12]	3	5,269	659
ml	iNLTK Headlines[12]	3	5,036	630
mr	iNLTK Headlines[12]	3	9,672	1,210
ta	iNLTK Headlines[12]	3	5,346	669
te	ACTSA corpus[13]	3	4,328	541

Table 11: Statistics of publicly available datasets (N is the number of classes)

B *IndicGLUE* News Category Dataset

The *IndicGLUE* news category dataset is a collection of articles labeled with news categories. We used this dataset in the evaluation of word embeddings and language models. Table 12 provides the statistics of the dataset.

C *IndicGLUE* Datasets

We provide some additional statistics for the *IndicGLUE* dataset in Table 6.

[9] https://github.com/NirantK/hindi2vec/releases/tag/bbc-hindi-v0.1
[10] http://www.iitp.ac.in/ ai-nlp-ml/resources.html
[11] https://www.kaggle.com/csoham/classification-bengali-news-articles-indicnlp
[12] https://github.com/goru001/inltk
[13] https://github.com/NirantK/bharatNLP/releases

Lang	Classes	# Articles	
		Train	Test
pa	BIZ, ENT, POL, SPT	2,496	312
bn	ENT, SPT	11,200	1,400
or	BIZ, CRM, ENT, SPT	17,750	2,250
gu	BIZ, ENT, SPT	1,632	204
mr	ENT, STY, SPT	3,600	450
kn	ENT, STY, SPT	24,000	3,000
te	ENT, BIZ, SPT	19,200	2,400
ml	BIZ, ENT, SPT, TECH	4,800	600
ta	ENT, POL, SPT	7,200	900

Table 12: *IndicGLUE* News category dataset statistics. The following are the categories: entertainment: ENT, sports: SPT, business: BIZ, lifestyle; STY, techology: TECH, politics: POL, crime: CRM.

	Min	Max	Avg
Headline Prediction			
Article Length (in words)	12	448	154
Headline Length (in words)	2	47	8.9
Wikipedia Section-Title Prediction			
Section Length (in words)	9	9554	140
Title Length (in words)	1	82	2.2
News Category Classification			
Article Length (in words)	23	4649	205
Cloze-style QA			
Question Length (in words)	7	190	63
Cross-lingual Sentence Retrieval			
Number of Sent Pairs per Lang Pair	752	6463	4904

Table 13: Additional *IndicGLUE* statistics